Pattern Recognition 45 (2012) 2251–2265

Contents lists available at SciVerse ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Determining the number of clusters using information entropy for mixed data

Jiye Liang [a,*], Xingwang Zhao [a], Deyu Li [a], Fuyuan Cao [a], Chuangyin Dang [b]

[a] Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, School of Computer and Information Technology, Shanxi University, Taiyuan, 030006 Shanxi, China

[b] Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong, Hong Kong

ARTICLE INFO

Article history:
Received 2 June 2011
Received in revised form
11 December 2011
Accepted 15 December 2011
Available online 24 December 2011

Keywords:
Clustering
Mixed data
Number of clusters
Information entropy
Cluster validity index
k-Prototypes algorithm

ABSTRACT

In cluster analysis, one of the most challenging and difficult problems is the determination of the number of clusters in a data set, which is a basic input parameter for most clustering algorithms. To solve this problem, many algorithms have been proposed for either numerical or categorical data sets. However, these algorithms are not very effective for a mixed data set containing both numerical attributes and categorical attributes. To overcome this deficiency, a generalized mechanism is presented in this paper by integrating Rényi entropy and complement entropy together. The mechanism is able to uniformly characterize within-cluster entropy and between-cluster entropy and to identify the worst cluster in a mixed data set. In order to evaluate the clustering results for mixed data, an effective cluster validity index is also defined in this paper. Furthermore, by introducing a new dissimilarity measure into the *k*-prototypes algorithm, we develop an algorithm to determine the number of clusters in a mixed data set. The performance of the algorithm has been studied on several synthetic and real world data sets. The comparisons with other clustering algorithms show that the proposed algorithm is more effective in detecting the optimal number of clusters and generates better clustering results.

1. Introduction

Clustering is an important tool in data mining, which has many applications in areas such as bioinformatics, web data analysis, information retrieval, customer relationship management, text mining, and scientific data exploration. It aims to partition a finite, unlabeled data set into several natural subsets so that data objects within the same clusters are close to each other and the data objects from different clusters are dissimilar from each other according to the predefined similarity measurement [1]. To accomplish this objective, many clustering algorithms have been proposed in the literature. For example, a detailed review of clustering algorithms and their applications can be found in [2–4]; clustering algorithms for high dimensional data are investigated in [5,6]; time series data clustering is reviewed in [7]; the clustering problem in the data stream domain is studied in [8,9]; and an overview of the approaches to clustering mixed data is given in [10].

At the very high end of the overall taxonomy, two main categories of clustering, known as partitional clustering and hierarchical clustering, are envisioned in the literature. The taxonomy of different clustering algorithms including state-of-the-art methods is depicted in Fig. 1. Most of these algorithms need a user-specified number of clusters or implicit cluster-number control parameters in advance. For some applications, the number of clusters can be estimated in terms of the user's expertise or domain knowledge. However, in many situations, the number of clusters for a given data set is unknown in advance. It is well known that over-estimation or under-estimation of the number of clusters will considerably affect the quality of clustering results. Therefore, identifying the number of clusters in a data set (a quantity often labeled k) is a fundamental issue in clustering analysis. To estimate the value of k, many studies have been reported in the literature [25]. Based on the differences in data types, these methods can be generally classified as clustering algorithms for numerical data, categorical data and mixed data.

In the numerical domain, Sun et al. [26] gave an algorithm based on the fuzzy k-means to automatically determine the number of clusters. It consists of a series of fuzzy k-means clustering procedures with the number of clusters varying from 2 to a predetermined k_{max}. By calculating the validity indices of the clustering results with different values of k ($2 \le k \le k_{max}$), the

* Corresponding author. Tel.: +86 3517010566.
E-mail addresses: ljy@sxu.edu.cn (J. Liang), zhaoxw84@163.com (X. Zhao), lidy@sxu.edu.cn (D. Li), cfy@sxu.edu.cn (F. Cao), mecdang@cityu.edu.hk (C. Dang).

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.12.017

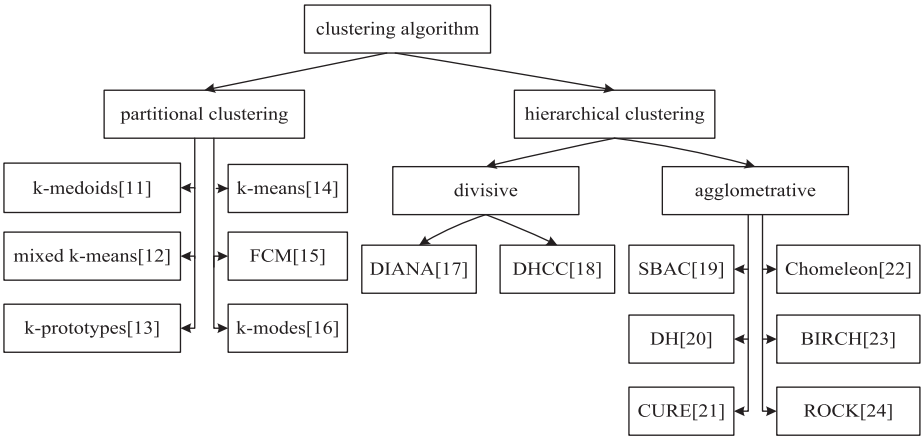

Fig. 1. The taxonomy of different clustering algorithms [11–24].

exact number of clusters in a given data set is obtained. Kothari et al. [27] presented a scale-based method for determining the number of clusters, in which the neighborhood serves as the scale parameter allowing for identification of the number of clusters based on persistence across a range of the scale parameter. Li et al. [28] presented an agglomerative fuzzy k-means clustering algorithm by introducing a penalty term to the objective function. Combined with cluster validation techniques, the algorithm can determine the number of clusters by analyzing the penalty factor. This method can find initial cluster centers and the number of clusters simultaneously. However, like the methods in [26,27], these approaches need implicit assumptions on the shape of the clusters characterized by distances to the centers of the clusters. Leung et al. [29] proposed an interesting hierarchical clustering algorithm based on human visual system research, in which each data point is regarded as a light point in an image, and a cluster is represented as a blob. As the real cluster should be perceivable over a wide range of scales, the lifetime of a cluster is used to test the "goodness" of a cluster and determine the number of clusters in a specific pattern of clustering. This approach focuses on the perception of human eyes and the data structure, which provides a new perspective for determining the number of clusters. Bandyopadhyay et al. [30–32] adopted the concept of variable length chromosome in genetic algorithm to tackle the issue of the unknown number of clusters in clustering algorithms. Other than evaluating the static clusters generated by a specific clustering algorithm, the validity functions in these approaches are used as clustering objective functions for computing the fitness, which guides the evolution to automatically search for a proper number of clusters from a given data set. Recently, information theory has been applied to determine the number of clusters. Sugar et al. [33] developed a simple yet powerful nonparametric method for choosing the number of clusters, whose strategy is to generate a distortion curve for the input data by running a standard clustering algorithm such as k-means for all values of k between 1 and n (the number of objects). The distortion curve, when transformed to an appropriate negative power, will exhibit a sharp jump at the "true" number of clusters, with the largest jump representing the best choice. Aghagoazadeh et al.[34] proposed a method for finding the number of clusters, which starts from a large number of clusters and reduces one cluster at each iteration and then allocates its data points to the remaining clusters. Finally, by measuring information potential, the exact number of clusters in a desired data set is determined.

For categorical data, Bai et al. [35] proposed an initialization method to simultaneously find initial cluster centers and the number of clusters. In this method, the candidates for the number of clusters can be obtained by comparing the possibility of the every initial cluster centers selected according to the density measure and the distance measure. Recently, a hierarchical entropy-based algorithm ACE (Agglomerative Categorical clustering with Entropy criterion) has been proposed in [36] for identifying the best ks, whose main idea is to find the best ks by observing the entropy difference between the neighboring clustering results, respectively. However, the complexity of this algorithm is proportional to the square of the number of objects. Transactional data is a kind of special categorical data, which can be transformed to the traditional row-by-column table with Boolean values. The ACE method becomes very time-consuming when applied to the transactional data, because the transactional data has two features: large volume and high dimensionality. In order to meet these potential challenges, based on the transactional-cluster-modes dissimilarity, Yan et al. [37] presented an agglomerative hierarchical transactional-clustering algorithm, which generates the merging dissimilarity indexes in hierarchical cluster merging processes. These indexes are used to find the candidate optimal number ks of clusters of transactional data.

In a real data set, it is more common to see both numerical attributes and categorical attributes at the same time. In other words, data are in a mixed mode. Over half of the data sets in the UCI Machine Learning Database Repository [64] are mixed data sets. For example, the Adult data set in the UCI Machine Learning Database Repository contains six numerical variables and eight categorical variables. There are several algorithms to cluster mixed data in the literature [12,13,19,20]. However, all these algorithms need to specify the number of clusters directly or indirectly in advance. Therefore, it still remains a challenging issue to determine the number of clusters in a mixed data set.

This paper aims to develop an effective method for determining the number of clusters in a given mixed data set. The method consists of a series of the modified k-prototypes procedures with the number of clusters varying from k_{max} to k_{min}, which results in a suite of successive clustering results. Concretely speaking, at each loop, basic steps of the method include: (1) Partitioning the input data set into the desired clusters utilizing the modified k-prototypes algorithm with a new defined dissimilarity measure, (2) evaluating the clustering results based on a proposed cluster validity index, (3) finding the worst cluster among these clusters using a generalized mechanism based on information entropy and then allocating the objects in this cluster into the remaining clusters using the dissimilarity measure, which reduces the overall number of clusters by one. At the beginning, the k_{max} cluster centers are randomly chosen. When the number of clusters decreases from $(k_{max}-1)$ to k_{min}, the cluster centers of

the current loop are obtained from the clustering results of the last loop. Finally, the plot of the cluster validity index versus the number of clusters for the given data is drawn. According to the plot, visual inspections can provide the optimal number of clusters for the given mixed data set. Experimental results on several synthetic and real data sets demonstrate the effectiveness of the method for determining the optimal number of clusters and obtaining better clustering results.

The remainder of the paper is organized as follows. In Section 2, a generalized mechanism is given. Section 3 presents an effective cluster validity index. Section 4 describes a modified *k*-prototypes algorithm and an algorithm for determining the number of clusters in a mixed data set. The effectiveness of the proposed algorithm is demonstrated in Section 5. Finally, concluding remarks are given in Section 6.

2. A generalized mechanism for mixed data

In the real world, many data sets are mixed-data sets, which consist of both numerical attributes and categorical attributes. In order to deal with mixed data in a uniform manner, a general strategy is to convert either categorical attributes into numerical attributes or numerical attributes into categorical attributes. However, this strategy has some drawbacks. On one hand, it is very difficult to assign correct numerical values to categorical values. For example, if color attribute takes values in the set {red, blue, green}, then one can convert the set into a numerical set such as {1, 2, 3}. Given this coding process, it will be inappropriate to compute distances between any coded values. On the other hand, to convert numerical into categorical, a discretizing algorithm has to be used to partition the value domain of a real-valued variable into several intervals and assign a symbol to all the values in the same interval. This process usually results in information loss since the membership degree of a value to discretized values is not taken into account [38]. Furthermore, the effectiveness of a clustering algorithm depends significantly on an underlying discretizing method. Therefore, it is desirable to develop a uniform computational method for directly clustering mixed data. In this section, based on information entropy, a generalized mechanism is presented for mixed data, which can be applied to characterize within-cluster entropy and between-cluster entropy and to identify the worst cluster of mixed data.

In general, mixed data are assumed to be stored in a table, where each row (tuple) represents facts about an object. Objects in the real world are usually characterized by both numerical attributes and categorical attributes at the same time. More formally, a mixed data table is described by a quadruple $MDT=(U,A,V,f)$, where:

(1) U is a nonempty set of objects, called a universe;
(2) A is a nonempty set of attributes with $A=A^r\cup A^c$, where A^r is a numerical attribute set and A^c is a categorical attribute set;
(3) V is the union of attribute domains, i.e., $V=\bigcup_{a\in A}V_a$, where V_a is the value domain of attribute a;
(4) $f: U\times A\to V$ is an information function such that, for any $a\in A$ and $x\in U$, $f(x,a)\in V_a$.

For convenience, a mixed data table $MDT=(U,A,V,f)$ is also denoted as $NDT=(U,A^r,V,f)$ and $CDT=(U,A^c,V,f)$, where $A=A^r\cup A^c$. *NDT* and *CDT* are called a numerical data table and a categorical data table, respectively.

Entropy is often used to measure the out-of-order degree of a system. The bigger the entropy value is, the higher the out-of-order degree of a system. The entropy of a system as defined by Shannon gives a measure of uncertainty about its actual structure [39]. It is a useful mechanism for characterizing the information content and has been used in a variety of applications including clustering [40], outlier detection [41], and uncertainty measure [42]. As follows, the entropy is extended to obtain a generalized mechanism for handling numerical data and categorical data uniformly. Owing to the difference in data types, information entropies for numerical data and categorical data will be introduced in the following, respectively.

For numerical data, Hungarian mathematician Alfred Rényi proposed a new information measure in the 1960s, named Rényi entropy [43]. It is the most general definition of information measures that preserve the additivity for independent events and can be directly estimated from data in a nonparametric fashion. The Rényi entropy for a stochastic variable x with probability density function $f(x)$ is defined as:

$$H_R(x)=\frac{1}{1-\alpha}\log\int f^{\alpha}(x)\,dx,\ \alpha>0,\quad \alpha\neq 1. \tag{1}$$

Specially, for $\alpha=2$, we obtain

$$H_R(x)=-\log\int f^2(x)\,dx, \tag{2}$$

which is called Rényi quadratic entropy.

In order to use Eq. (2) in the calculations, we need a way to estimate the probability density function. One of the most productive nonparametric methods is the Parzen window density estimation [44], which is a well-known kernel-based density estimation method. Given a set of independent identical distribution samples $\{x_1,x_2,\ldots,x_N\}$ with d numerical attributes drawn from the true density $f(x)$, the Parzen window estimator for this distribution is defined as:

$$\hat{f}(x)=\frac{1}{N}\sum_{i=1}^{N}W_{\sigma^2}(x,x_i). \tag{3}$$

Here, W_{σ^2} is the Parzen window and σ^2 controls the width of the kernel. The Parzen window must integrate to one, and is typically chosen to be a probability distribution function such as the Gaussian kernel, i.e.,

$$W_{\sigma^2}(x,x_i)=\frac{1}{(2\pi)^{d/2}\sigma^d}\exp\left(-\frac{(x-x_i)^T(x-x_i)}{2\sigma^2}\right). \tag{4}$$

As a result, from the plug-in-a-density-estimator principle, we obtain an estimate for the Rényi entropy by replacing $f(x)$ with $\hat{f}(x)$. Since the logarithm is a monotonic function, we only need to focus on the quantity $V(f)=\int\hat{f}^2(x)\,dx$, which is given by

$$\begin{aligned}V(f)&=\int\frac{1}{N}\sum_{i=1}^{N}W_{\sigma^2}(x,x_i)\frac{1}{N}\sum_{j=1}^{N}W_{\sigma^2}(x,x_j)\,dx\\&=\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\int W_{\sigma^2}(x,x_i)W_{\sigma^2}(x,x_j)\,dx.\end{aligned} \tag{5}$$

By the convolution theorem for Gaussians [45], we have

$$\int W_{\sigma^2}(x,x_i)W_{\sigma^2}(x,x_j)\,dx=W_{2\sigma^2}(x_i,x_j). \tag{6}$$

That is, the convolution of two Gaussians is a new Gaussian function having twice the covariance. Thus,

$$V(f)=\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}W_{2\sigma^2}(x_i,x_j). \tag{7}$$

To make sure the Rényi entropy is positive, the Gaussian function $W_{2\sigma^2}$ can be multiplied by a sufficiently small positive number β so that $W'_{2\sigma^2}=\beta\times W_{2\sigma^2}$. In the following, the Within-cluster Entropy (abbreviated as *WE_N*), Between-cluster Entropy (abbreviated as *BE_N*) and Sum of Between-cluster Entropies in

Absence of a cluster (abbreviated as *SBAE_N*) for Numerical data are defined based on the above analysis, respectively.

The within-cluster entropy for numerical data is given as follows [43].

Definition 1. Let $NDT=(U,A^r,V,f)$ be a numerical data table, which can be separated into k clusters, i.e., $C^k=\{C_1,C_2,\ldots,C_k\}$. For any $C_{k'}\in C^k$, the $WE\_N(C_{k'})$ is defined as:

$$WE\_N(C_{k'})=-\log\frac{1}{N_{k'}^2}\sum_{x\in C_{k'}}\sum_{y\in C_{k'}}W'_{2\sigma^2}(x,y), \tag{8}$$

where $N_{k'}=|C_{k'}|$ is the number of objects in the cluster $C_{k'}$.

In order to evaluate the difference between clusters, the between-cluster entropy for numerical data, which was first introduced by Gockay et al. [46], is defined as follows.

Definition 2. Let $NDT=(U,A^r,V,f)$ be a numerical data table, which can be separated into k clusters, i.e., $C^k=\{C_1,C_2,\ldots,C_k\}$. For any $C_i,C_j\in C^k(i\neq j)$, the $BE\_N(C_i,C_j)$ is defined as:

$$BE\_N(C_i,C_j)=-\log\frac{1}{N_iN_j}\sum_{x\in C_i}\sum_{y\in C_j}W'_{2\sigma^2}(x,y), \tag{9}$$

where $N_i=|C_i|$ and $N_j=|C_j|$ represent the number of objects in the clusters C_i and C_j, respectively.

Intuitively, if two clusters are well separated, the *BE_N* will have a relatively large value. This provides us with a tool for cluster evaluation. Furthermore, in order to characterize the effect of a cluster on the clustering results, the sum of between-cluster entropies in absence of a cluster for a numerical data set [34] is described as follows.

Definition 3. Let $NDT=(U,A^r,V,f)$ be a numerical data table, which can be separated into $k(k>2)$ clusters, i.e., $C^k=\{C_1,C_2,\ldots,C_k\}$. For any $C_{k'}\in C^k$, the $SBAE\_N(C_{k'})$ is defined as:

$$SBAE\_N(C_{k'})=\sum_{C_i\in C^k,i\neq k'}\sum_{C_j\in C^k,j\neq k',j\neq i}BE\_N(C_i,C_j). \tag{10}$$

Obviously, the larger the $SBAE\_N(C_{k'})$ is, the less the effect of the cluster $C_{k'}$ on the clustering results. That is to say, if the $SBAE\_N(C_{k'})$ is the largest, the clustering results excluding the cluster $C_{k'}$ will be the best.

In a categorical domain, Liang et al. [47] used the complement entropy to measure information content and uncertainty for a categorical data table. Unlike the logarithmic behavior of Shannon's entropy, the complement entropy can measure both uncertainty and fuzziness. Recently, it has been used in a variety of applications for categorical data including feature selection [48], rule evaluation [49], and uncertainty measure [47,50,51].

Definition 4. Let $CDT=(U,A^c,V,f)$ be a categorical data table and $P\subseteq A^c$. A binary relation $IND(P)$, called indiscernibility relation, is defined as

$$IND(P)=\{(x,y)\in U\times U\,|\,\forall a\in P,f(x,a)=f(y,a)\}. \tag{11}$$

Two objects are indiscernible in the context of a set of attributes if they have the same values for those attributes. $IND(P)$ is an equivalence relation on U and $IND(P)=\bigcap_{a\in P}IND(\{a\})$.

The relation $IND(P)$ induces a partition of U, denoted by $U/IND(P)=\{[x]_P\,|\,x\in U\}$, where $[x]_P$ denotes the equivalence class determined by x with respect to P, i.e., $[x]_P=\{y\in U\,|\,(x,y)\in IND(P)\}$.

The complement entropy for categorical data is defined as follows [47].

Definition 5. Let $CDT=(U,A^c,V,f)$ be a categorical data table, $P\subseteq A^c$ and $U/IND(P)=\{X_1,X_2,\ldots,X_m\}$. The complement entropy with respect to P is defined as

$$E(P)=\sum_{i=1}^{m}\frac{|X_i|}{|U|}\frac{|X_i^c|}{|U|}=\sum_{i=1}^{m}\frac{|X_i|}{|U|}\left(1-\frac{|X_i|}{|U|}\right), \tag{12}$$

where X_i^c denotes the complement set of X_i, i.e., $X_i^c=U-X_i$, $|X_i|/|U|$ represents the probability of X_i within the universe U and $|X_i^c|/|U|$ is the probability of the complement set of X_i within the universe U.

Based on the complement entropy, the Within-cluster Entropy (abbreviated as *WE_C*), Between-cluster Entropy (abbreviated as *BE_C*) and Sum of Between-cluster Entropies in Absence of a cluster (abbreviated as *SBAE_C*) for Categorical data are defined as follows.

Definition 6. Let $CDT=(U,A^c,V,f)$ be a categorical data table, which can be separated into k clusters, i.e., $C^k=\{C_1,C_2,\ldots,C_k\}$. For any $C_{k'}\in C^k$, the $WE\_C(C_{k'})$ is defined as

$$WE\_C(C_{k'})=\sum_{a\in A^c}\sum_{X\in C_{k'}/IND(\{a\})}\frac{|X|}{|C_{k'}|}\left(1-\frac{|X|}{|C_{k'}|}\right). \tag{13}$$

Definition 7 (*Huang [13]*). Let $CDT=(U,A^c,V,f)$ be a categorical data table. For any $x,y\in U$, the dissimilarity measure $D_{A^c}(x,y)$ is defined as

$$D_{A^c}(x,y)=\sum_{a\in A^c}d_a(x,y), \tag{14}$$

where

$$d_a(x,y)=\begin{cases}0, & f(x,a)=f(y,a),\\ 1, & f(x,a)\neq f(y,a).\end{cases}$$

Intuitively, the dissimilarity between two categorical objects is directly proportional to the number of attributes in which they mismatch. Furthermore, we find there is a quantitative relation between $WE\_C(C_{k'})$ and $D_{A^c}(x,y)$, i.e.,

$$WE\_C(C_{k'})=\frac{1}{|C_{k'}|^2}\sum_{x\in C_{k'}}\sum_{y\in C_{k'}}D_{A^c}(x,y), \tag{15}$$

which is proved as follows.

For convenience, suppose that $Y_a=C_{k'}/IND(\{a\})$, where $a\in A^c$. Then,

$$\begin{aligned}WE\_C(C_{k'})&=\sum_{a\in A^c}\sum_{X\in Y_a}\frac{|X|}{|C_{k'}|}\left(1-\frac{|X|}{|C_{k'}|}\right)\\&=\sum_{a\in A^c}\left(1-\sum_{X\in Y_a}\frac{|X|^2}{|C_{k'}|^2}\right)\\&=\frac{1}{|C_{k'}|^2}\sum_{a\in A^c}\left(|C_{k'}|^2-\sum_{X\in Y_a}|X|^2\right)\\&=\frac{1}{|C_{k'}|^2}\sum_{a\in A^c}\sum_{x\in C_{k'}}\sum_{y\in C_{k'}}d_a(x,y)\\&=\frac{1}{|C_{k'}|^2}\sum_{x\in C_{k'}}\sum_{y\in C_{k'}}\sum_{a\in A^c}d_a(x,y)\\&=\frac{1}{|C_{k'}|^2}\sum_{x\in C_{k'}}\sum_{y\in C_{k'}}D_{A^c}(x,y).\end{aligned}$$

The above derivation means that the within-cluster entropy can be expressed with the average dissimilarity between objects within a cluster for categorical data. Therefore, based on the

average dissimilarity between pairs of samples in two different clusters, the between-cluster entropy for categorical data is defined as follows.

Definition 8. Let $CDT=(U,A^c,V,f)$ be a categorical data table, which can be separated into k clusters, i.e., $C^k=\{C_1,C_2,\ldots,C_k\}$. For any $C_i,C_j\in C^k(i\neq j)$, the $BE\_C(C_i,C_j)$ is defined as:

$$BE\_C(C_i,C_j)=\frac{1}{N_iN_j}\sum_{x\in C_i}\sum_{y\in C_j}D_{A^c}(x,y), \tag{16}$$

where $N_i=|C_i|$ and $N_j=|C_j|$.

Given this definition, we obtain the following entropy.

Definition 9. Let $CDT=(U,A^c,V,f)$ be a categorical data table, which can be separated into $k(k>2)$ clusters, i.e., $C^k=\{C_1,C_2,\ldots,C_k\}$. For any $C_{k'}\in C^k$, the $SBAE\_C(C_{k'})$ is defined as:

$$SBAE\_C(C_{k'})=\sum_{C_i\in C^k,i\neq k'}\sum_{C_j\in C^k,j\neq k',j\neq i}BE\_C(C_i,C_j). \tag{17}$$

By integrating the *SBAE_N* and *SBAE_C* together, the Sum of Between-cluster Entropies in Absence of a cluster (abbreviated as *SBAE_M*) for Mixed data can be calculated as follows.

Definition 10. Let $MDT=(U,A,V,f)$ be a mixed data table, which can be separated into $k(k>2)$ clusters, i.e., $C^k=\{C_1,C_2,\ldots,C_k\}$. For any $C_{k'}\in C^k$, the $SBAE\_M(C_{k'})$ is defined as:

$$SBAE\_M(C_{k'})=\frac{|A^r|}{|A|}\frac{SBAE\_N(C_{k'})}{\sum_{i=1}^{k}SBAE\_N(C_i)}+\frac{|A^c|}{|A|}\frac{SBAE\_C(C_{k'})}{\sum_{i=1}^{k}SBAE\_C(C_i)}. \tag{18}$$

It is well known that the effect of different clusters on the clustering results is not equal. Since the best clustering is achieved when clusters have the maximum dissimilarity, hence, the larger the between-cluster entropy is, the better the clustering is. The cluster, without which the remaining clusters become the most separate clusters, is called the worst cluster. That is to say, this cluster has the smallest effect on the between-cluster entropy among all the clusters. Based on the *SBAE_M*, the definition of the worst cluster is as follows.

Definition 11. Let $MDT=(U,A,V,f)$ be a mixed data table, which can be separated into $k(k>2)$ clusters, i.e., $C^k=\{C_1,C_2,\ldots,C_k\}$. The worst cluster $C_w\in C^k$ is defined as:

$$C_w=\arg\max_{C_{k'}\in C^k}SBAE\_M(C_{k'}). \tag{19}$$

In the following, the process of identifying the worst cluster among the clustering results is illustrated in Example 1.

Example 1. Consider the artificial data set given in Table 1, where $U=\{x_1,x_2,\ldots,x_9\}$ and $A=A^c\cup A^r=\{a_1,a_2,a_3,a_4\}$, with $A^c=\{a_1,a_2\}$ and $A^r=\{a_3,a_4\}$. Let U be partitioned into three clusters $C^3=\{C_1,C_2,C_3\}$, where $C_1=\{x_1,x_2,x_3\}$, $C_2=\{x_4,x_5,x_6\}$ and $C_3=\{x_7,x_8,x_9\}$.

Suppose that the kernel size σ is set to 0.05 in the Gaussian kernel. According to Definition 3, the sum of between-cluster entropies in absence of a cluster for numerical attributes are given by

$$SBAE\_N(C_1)=4.3017,$$

$$SBAE\_N(C_2)=3.5520$$

Table 1
An artificial data set.

Objects	a_1	a_2	a_3	a_4	Clusters
x_1	a	f	0.50	0.60	
x_2	b	f	0.45	0.48	C_1
x_3	c	e	0.55	0.49	
x_4	b	e	0.30	0.35	
x_5	b	f	0.27	0.47	C_2
x_6	c	e	0.35	0.48	
x_7	a	f	0.52	0.32	
x_8	a	d	0.43	0.20	C_3
x_9	c	d	0.55	0.24	

and

$$SBAE\_N(C_3)=1.8141.$$

Similarly, according to Definition 9, the sum of between-cluster entropies in absence of a cluster for categorical attributes are given by

$$SBAE\_C(C_1)=\frac{16}{9},$$

$$SBAE\_C(C_2)=\frac{13}{9}$$

and

$$SBAE\_C(C_3)=\frac{11}{9}.$$

Finally, the sum of between-cluster entropies in absence of a cluster for mixed attributes are

$$\begin{aligned}SBAE\_M(C_1)&=\frac{2}{4}\times\frac{4.3017}{4.3017+3.5520+1.8141}\\&\quad+\frac{2}{4}\times\frac{16/9}{16/9+13/9+11/9}\\&=0.4225,\end{aligned}$$

$$\begin{aligned}SBAE\_M(C_2)&=\frac{2}{4}\times\frac{3.5520}{4.3017+3.5520+1.8141}\\&\quad+\frac{2}{4}\times\frac{13/9}{16/9+13/9+11/9}\\&=0.3462\end{aligned}$$

and

$$\begin{aligned}SBAE\_M(C_3)&=\frac{2}{4}\times\frac{1.8141}{4.3017+3.5520+1.8141}\\&\quad+\frac{2}{4}\times\frac{11/9}{16/9+13/9+11/9}\\&=0.2313.\end{aligned}$$

Obviously, $SBAE\_M(C_1)>SBAE\_M(C_2)>SBAE\_M(C_3)$. Thus, according to Definition 11, the cluster C_1 is the worst cluster.

3. Cluster validity index

To evaluate the clustering results, a number of cluster validity indices have been given in the literature [26,52–54]. However, these cluster validity indices are only applicable for either numerical data or categorical data. As follows, we propose an effective cluster validity index based on the category utility function introduced by Gluck and Corter [55]. The category utility

function is a measure of "category goodness", which has been applied in some clustering algorithms [56,57] and can be described as follows.

Suppose that a categorical data table $CDT=(U,A^c,V,f)$ has a partition $C^k=\{C_1,C_2,\ldots,C_k\}$ with k clusters, which are found by a clustering algorithm. Then the category utility function of the clustering results C^k for categorical data is calculated by [55]

$$CUC(C^k)=\frac{1}{k}\sum_{a\in A^c}Q_a, \tag{20}$$

where

$$Q_a=\sum_{X\in U/IND(\{a\})}\sum_{i=1}^{k}\frac{|C_i|}{|U|}\left(\frac{|X\cap C_i|^2}{|C_i|^2}-\frac{|X|^2}{|U|^2}\right).$$

One can see that the category utility function is defined in terms of the bivariate distributions of a clustering result and each of the features, which looks different from more traditional clustering criteria adhering to similarities and dissimilarities between instances. Mirkin [58] shows that the category utility function is equivalent to the square error criterion in traditional clustering, when a standard encoding scheme of categories is applied. As follows, a corresponding category utility function for numerical data is given [58].

Suppose that a numerical data table $NDT=(U,A^r,V,f)$ with $A^r=\{a_1,a_2,\ldots,a_{|A^r|}\}$ can be separated into k clusters, i.e., $C^k=\{C_1,C_2,\ldots,C_k\}$, by a clustering algorithm. Then the category utility function of the clustering results C^k for numerical data is defined by

$$CUN(C^k)=\frac{1}{k}\sum_{l=1}^{|A^r|}\left(\delta_l^2-\sum_{j=1}^{k}p_j\delta_{jl}^2\right), \tag{21}$$

where $\delta_l^2=\sum_{x\in U}(f(x,a_l)-m_l)^2/|U|$ and $\delta_{jl}^2=\sum_{x\in C_j}(f(x,a_l)-m_{jl})^2/|C_j|$ are the variance and within-class variance of the attribute a_l, respectively; m_l and m_{jl} denote the grand mean and within-class mean of the attribute a_l, respectively; and $p_j=|C_j|/|U|$.

Based on Eqs. (20) and (21), a validity index for the clustering results, i.e., $C^k=\{C_1,C_2,\ldots,C_k\}$, obtained by a clustering algorithm on the mixed data table $MDT=(U,A,V,f)$, is defined as:

$$CUM(C^k)=\frac{|A^r|}{|A|}CUN(C^k)+\frac{|A^c|}{|A|}CUC(C^k). \tag{22}$$

It is clear that the higher the value of *CUM* above, the better the clustering results. The cluster number which maximizes *CUM* is considered to be the optimal number of clusters in a mixed data set.

4. An algorithm for determining the number of clusters in a mixed data set

In this section, we first review the k-prototypes algorithm, and then redefine the dissimilarity measure used in the k-prototypes algorithm. Based on the generalized mechanism using information entropy, the proposed cluster validity index and the modified k-prototypes algorithm, an algorithm for determining the number of clusters in a mixed data set is proposed.

4.1. A modified k-prototypes algorithm

In 1998, Huang [13] proposed the k-prototypes algorithm, which is a simple integration of the k-means [14] and k-modes [16] algorithms. The k-prototypes algorithm is widely used because frequently encountered objects in real world database are mixed-type objects, and it is efficient in processing large data sets. In the k-prototypes algorithm, the dissimilarity measure takes into account both numerical attributes and categorical attributes. The dissimilarity measure on numerical attributes is defined by the squared Euclidean distance. For the categorical part, the computation of dissimilarity is performed by simple matching, which is the same as that of the k-modes. The dissimilarity between two mixed-type objects $x,y\in U$, can be measured by [13]

$$D(x,y)=D_{A^r}(x,y)+\gamma D_{A^c}(x,y), \tag{23}$$

where $D_{A^r}(x,y)$ and $D_{A^c}(x,y)$ represent the dissimilarities of the numerical and categorical parts, respectively. $D_{A^r}(x,y)$ is calculated according to

$$D_{A^r}(x,y)=\sum_{a\in A^r}(f(x,a)-f(y,a))^2. \tag{24}$$

$D_{A^c}(x,y)$ is calculated according to Eq. (14). The weight γ is used to control the relative contribution of numerical and categorical attributes when computing the dissimilarities between objects.

However, how to choose an appropriate γ is a very difficult problem in practice. To overcome this difficulty, we modify the $D(x,y)$. A new dissimilarity between two mixed-type objects $x,y\in U$ is given as follows:

$$D(x,y)=\frac{|A^r|}{|A|}D_{A^r}(x,y)+\frac{|A^c|}{|A|}D_{A^c}(x,y). \tag{25}$$

As a matter of fact, the dissimilarity used in k-prototypes algorithm is calculated between an object and a prototype. And the ranges of dissimilarity measures for numerical attributes and categorical attributes are different. In order to reflect the relative contributions of numerical and categorical attributes, we modify $D(x,y)$ in the following way.

Suppose that the clustering results of a mixed data table $MDT=(U,A,V,f)$ are $C^k=\{C_1,C_2,\ldots,C_k\}$, whose cluster prototypes are $Z^k=\{z_1,z_2,\ldots,z_k\}$, where k is the number of clusters. The dissimilarity between $x\in U$ and the prototype $z\in Z^k$, is measured by

$$D(x,z)=\frac{|A^r|}{|A|}\frac{D_{A^r}(x,z)}{\sum_{i=1}^{k}D_{A^r}(x,z_i)}+\frac{|A^c|}{|A|}\frac{D_{A^c}(x,z)}{\sum_{i=1}^{k}D_{A^c}(x,z_i)}, \tag{26}$$

where $D_{A^r}(x,z)$ and $D_{A^c}(x,z)$ are calculated according to Eqs. (24) and (14), respectively.

Based on this dissimilarity, a modified k-prototypes algorithm is proposed, which is as follows.

Step 1: Choose k distinct objects from the mixed data table $MDT=(U,A,V,f)$ as the initial prototypes.

Step 2: Allocate each object in $MDT=(U,A,V,f)$ to a cluster whose prototype is the nearest to it according to Eq. (26). Update the prototypes after each allocation.

Step 3: After all objects have been allocated to clusters, recalculate the similarity of objects against the current prototypes. If an object is found such that its nearest prototype belongs to another cluster rather than its current one, reallocate the object to that cluster and update the corresponding prototypes.

Step 4: Repeat Step 3 till no object changes from one cluster to another or a given stopping criterion is fulfilled.

To better understand the modified k-prototypes algorithm, iterations of this algorithm are illustrated in Example 2.

Example 2 (*Continued from Example 1*). Suppose that the initial prototypes are $\{x_1,x_4,x_7\}$. According to Eq. (26), the dissimilarity between each object of $U=\{x_1,x_2,\ldots,x_9\}$ and the prototypes is shown in Table 2. Furthermore, executing the Step 2 of the modified k-prototypes algorithm, we obtain three clusters, i.e.,

Table 2
The dissimilarity between each object of U and the prototypes.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
x_1	0	0.2494	0.2441	0.5456	0.3811	0.3807	0.2214	0.4546	0.4050
x_4	0.8227	0.4330	0.4771	0	0.1946	0.1659	0.7786	0.3605	0.4138
x_7	0.1773	0.3176	0.2788	0.4544	0.4244	0.4534	0	0.1850	0.1812

Table 3
The algorithm for determining the number of clusters in a mixed data set.

Input: A mixed data table $MDT=(U,A,V,f)$, k_{min}, k_{max}
and the kernel size σ.
Arbitrarily choose k_{max} objects $z_1,z_2,\ldots,z_{k_{max}}$ from the mixed data table MDT as the initial cluster centers $Z^{k_{max}}=\{z_1,z_2,\ldots,z_{k_{max}}\}$.
For $i=k_{max}$ to k_{min}
  Apply the modified k-prototypes with the initial centers Z^i on the mixed data table MDT and return the clustering results $C^i=\{C_1,C_2,\ldots,C_i\}$;
  According to Eq. (22), compute the cluster validity index $CUM(C^i)$ for the clustering results C^i;
  According to Eq. (19), identify the worst cluster C_w, $C_w\in C^i$;
  For any $x\in C_w$, assign x to an appropriate cluster based on the minimum of dissimilarity measure using Eq. (26);
  Update the centers of clusters, which are used as the expected centers of clusters for the next loop;
End;
Compare the validity indices and choose k such that
$k=\arg\max_{i=k_{max},\ldots,k_{min}} CUM(C^i)$;
Output: The optimal number of clusters k.

$C_1=\{x_1,x_2,x_3\}$, $C_2=\{x_4,x_5,x_6\}$ and $C_3=\{x_7,x_8,x_9\}$, and the corresponding cluster prototypes are $z_1=\{a,f,0.5,0.5233\}$, $z_2=\{b,e,0.3067,0.4333\}$ and $z_3=\{a,d,0.5,0.2533\}$ in the current iteration process, respectively.

4.2. Overview of the proposed algorithm

Based on the above mentioned formulations and notation, an algorithm is developed for determining the number of clusters in mixed data, which is described in Table 3.

Referring to the proposed algorithm, the time complexity is analyzed as follows. In each loop, the time complexity mainly consists of two parts. In the first part, the cost of applying the modified k-prototypes algorithm on the input data set to obtain i clusters is $O(it|U||A|)$, where t is the number of iterations of the modified k-prototypes algorithm in current loop. On the other hand, when identifying the worst cluster, the between-cluster entropy needs to be calculated between any pair of clusters, and thus the time complexity of this calculation is $O(|U|^2|A|^2)$. Therefore, the overall time complexity of the proposed algorithm is $O((k_{max}-k_{min}+1)(|U|^2|A|^2))$.

5. Experimental analysis

In this section, we evaluate the effectiveness of the proposed algorithm in detecting the optimal number of clusters and obtaining better clustering results. We have carried out a number of experiments on both synthetic and real data sets. On the one hand, in order to evaluate the ability of detecting the optimal number of clusters, the proposed algorithm was compared with the method in [59]. On the other hand, the comparisons between the proposed algorithm and the other algorithms with a known number of clusters (the modified k-prototypes algorithm and k-centers algorithm [60]) have been implemented to evaluate the effectiveness of obtaining better clustering results. In the following experiments, unless otherwise mentioned, the kernel size σ in the proposed algorithm is set to 0.05. And the weight parameter γ used in the k-centers algorithm [60] is set to 0.5. To avoid the influence of the randomness arising from the initialization of cluster centers, each experiment is executed 100 times on the same data set. As choosing the best range of the number of clusters is a difficult problem, we have adopted Bezdek's suggestion of $k_{min}=2$ and $k_{max}=\sqrt{n}$ [61], where n is the number of objects in the data set. To evaluate the results of clustering algorithms, two criteria are introduced in the following.

Table 4
Contingency table for comparing partitions P and Q.

Partition	Group	Q				Total
		q_1	q_2	$\cdots$	q_c	
P	p_1	t_{11}	t_{12}	$\cdots$	t_{1c}	$t_{1.}$
	p_2	t_{21}	t_{22}	$\cdots$	t_{2c}	$t_{2.}$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
	p_k	t_{k1}	t_{k2}	$\cdots$	t_{kc}	$t_{k.}$
Total		$t_{.1}$	$t_{.2}$	$\cdots$	$t_{.c}$	$t_{..}=n$

Table 5
Means and variances of the ten-cluster data set.

Attributes	Mean – Variance	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}
X	Mean	0.0	5.0	5.0	1.5	−2.0	−5.5	−5	−2.0	2.0	7.0
	Variance	0.8	0.5	0.5	0.8	1.0	1.0	0.5	0.5	0.5	1.0
Y	Mean	0.0	4.5	0.0	3.0	−4.5	−1.5	2.0	4.5	−3.5	−3.5
	Variance	0.8	0.5	0.5	0.5	1.0	0.5	1.0	1.0	1.0	0.5

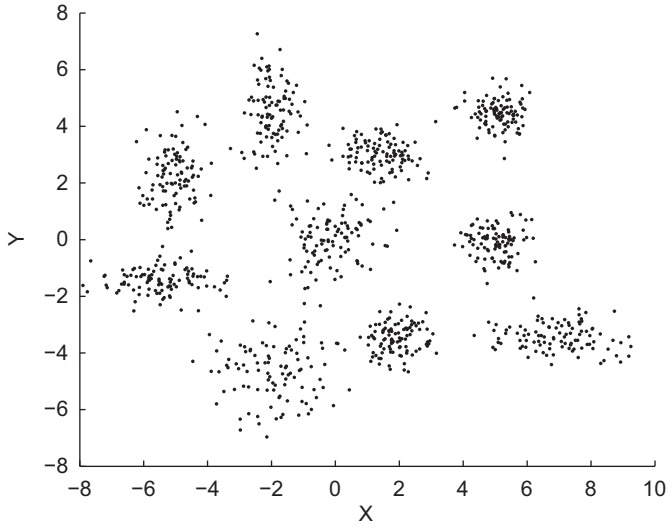

Fig. 2. Scatter plot of the ten-cluster data set.

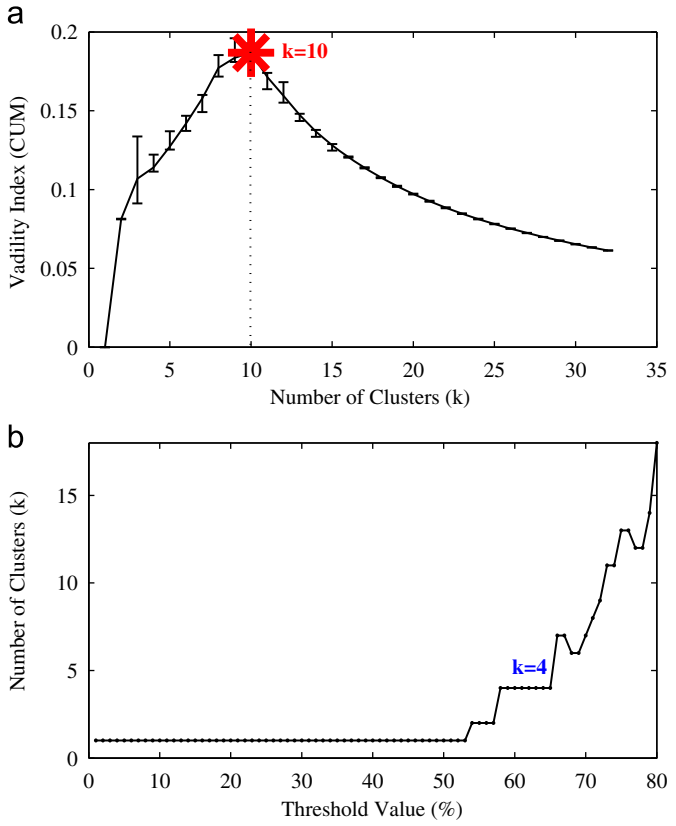

Fig. 3. The estimated number of clusters for the ten-cluster data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

- *Category utility function for mixed data*: The category utility function for mixed data (abbreviated as *CUM*) is an internal criterion which attempts to maximize both the probability that two data objects in the same cluster have attribute values in common and the probability that data points from different clusters have different values. The formula for calculating the expected value of the *CUM* can be found in Section 3.
- *Adjusted rand index*: The adjusted rand index [62], also referred to as *ARI*, is a measure of agreement between two partitions: one given by a clustering algorithm and the other defined by external criteria. Consider a set of n objects $U=\{x_1,x_2,\ldots,x_n\}$ and suppose that $P=\{p_1,p_2,\ldots,p_k\}$ and $Q=\{q_1,q_2,\ldots,q_c\}$ represent two different partitions of the objects in U such that $\bigcup_{i=1}^{k} p_i=\bigcup_{j=1}^{c} q_j=U$ and $p_i\cap p_{i'}=q_j\cap q_{j'}=\emptyset$ for $1\le i\ne i'\le k$ and $1\le j\ne j'\le c$. Given two partitions, P and Q, with k and c subsets, respectively, the contingency table (see Table 4) can be formed to indicate group overlap between P and Q.

In Table 4, a generic entry, t_{ij}, represents the number of objects that were classified in the ith subset of partition P and in the jth subset of partition Q. *ARI* can be computed by

$$ARI=\frac{\binom{n}{2}\sum_{i=1}^{k}\sum_{j=1}^{c}\binom{t_{ij}}{2}-\left[\sum_{i=1}^{k}\binom{t_i}{2}\sum_{j=1}^{c}\binom{t_j}{2}\right]}{\frac{1}{2}\binom{n}{2}\left[\sum_{i=1}^{k}\binom{t_i}{2}+\sum_{j=1}^{c}\binom{t_j}{2}\right]-\left[\sum_{i=1}^{k}\binom{t_i}{2}\sum_{j=1}^{c}\binom{t_j}{2}\right]} \quad (27)$$

with maximum value 1. If the clustering result is close to the true class distribution, then the value of *ARI* is high.

5.1. Numerical examples with synthetic data sets

The 1000 synthetic numerical data points were generated from a mixture of Gaussian distributions with 10 clusters (also referred to as ten-cluster data set). Each data point was described by two numerical attributes (X and Y). These attribute values were generated by sampling normal distributions with different means and variances for each cluster. The means and variances of the ten clusters are given in Table 5. The scatter plot of this generated data set is shown in Fig. 2. Fig. 3 shows the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm given in [59]. Table 6 lists the results of three different algorithms on this data set.

We have also examined the result on the synthetic data set with numerical attributes and categorical attributes, which was used in [20]. The data set, named student, has 600 objects with

Table 6
The summary results of three algorithms on the ten-cluster data set.

Indices	Modified k-Prototypes	k-Centers	Proposed algorithm
CUM	0.1857	0.1837	**0.1868**
ARI	0.7252	0.7679	**0.8270**

Table 7
Synthetic mixed student data set.

Sex	Age	Amount	Product	Department	College
M(50%),FM(50%)	$N(20,2)$	70–90	Orange	I.D.	Design
			Apple	V.C.	
			Coke	E.E.	Engineering
			Pepsi	M.E.	
			Rice	I.S.	Management
			Bread	B.A.	

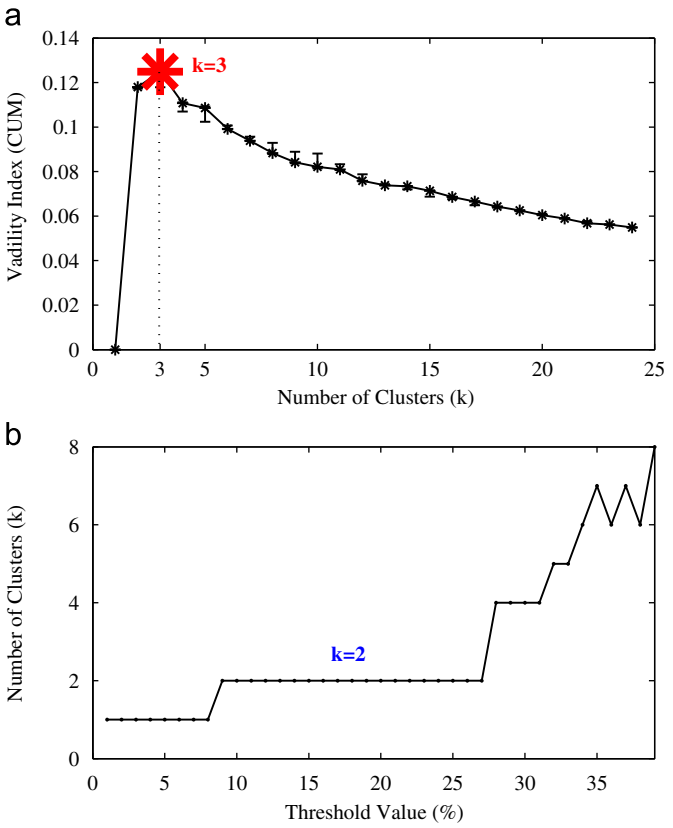

Fig. 4. The estimated number of clusters for the student data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

Table 8
The summary results of three algorithms on the student data set.

Indices	Modified *k*-prototypes	*k*-Centers	Proposed algorithm
CUM	0.1168	0.1254	**0.1256**
ARI	0.5063	**0.8120**	0.5362

Table 9
The summary of real data sets' characteristics.

Data sets	Abbreviation	# Instances	# Numerical attributes	# Categorical attributes	# Clusters
Wine recognition	Wine	178	13	0	3
Wisconsin Breast Cancer	Breast cancer	699	9	0	2
Congressional voting records	Voting	435	0	16	2
Car evaluation database	Car	1728	0	6	4
Splice-junction gene sequences	DNA	3190	0	60	3
Teaching assistant evaluation	TAE	151	1	4	3
Heart disease	Heart	303	5	8	2
Australian credit approval	Credit	690	6	8	2
Contraceptive method choice	CMC	1473	2	7	3
Adult	Adult	44 842	6	8	2

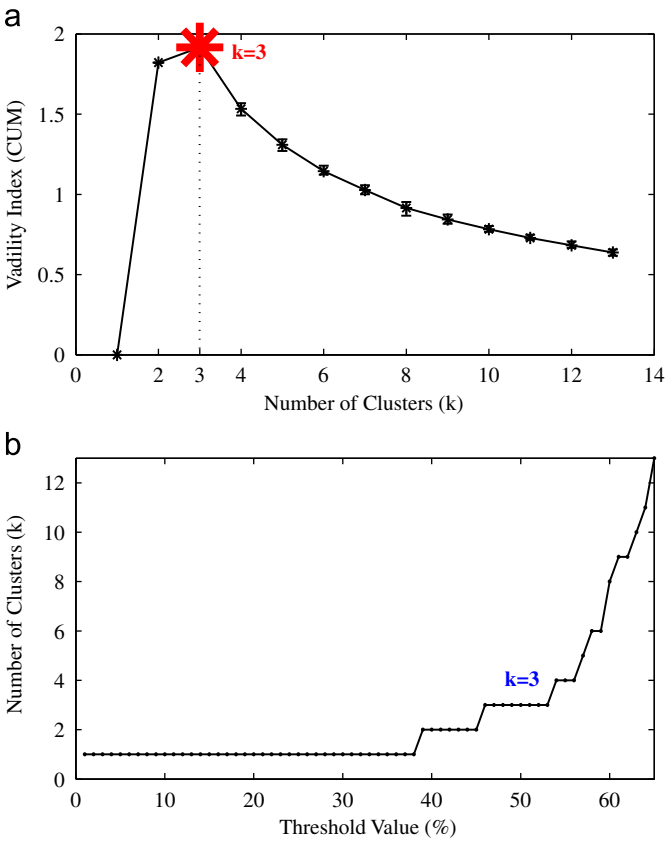

Fig. 5. The estimated number of clusters for the Wine data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

Table 10
The summary results of three algorithms on the Wine data set.

Indices	Modified *k*-prototypes	*k*-Centers	Proposed algorithm
CUM	1.8714	1.8834	**1.9166**
ARI	0.8025	0.8076	**0.8471**

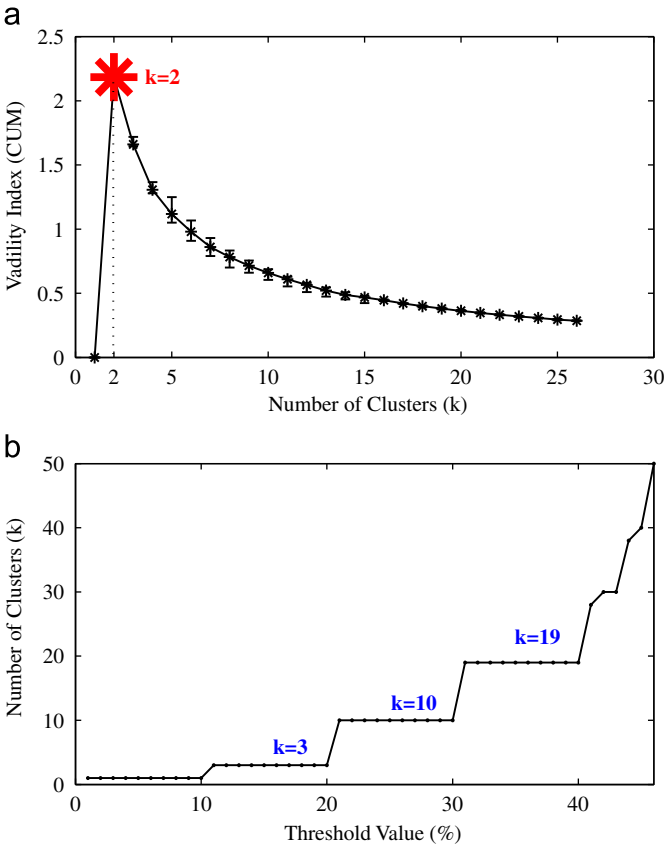

Fig. 6. The estimated number of clusters for the Breast Cancer data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

Table 11
The summary results of three algorithms on the Breast Cancer data set.

Indices	Modified *k*-prototypes	*k*-Centers	Proposed algorithm
CUM	2.1490	2.1210	**2.1840**
ARI	0.8040	0.7967	**0.8216**

data distribution as shown in Table 7. The data has six attributes: three categorical attributes (sex, product, and department), two numerical attributes (age and amount), and one decision attribute (college). The latter does not participate in clustering. The class value of each pattern is assigned deliberately according to its department and product values to facilitate the measurement of cluster quality. Fig. 4 shows the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm in [59]. The summary results of three different algorithms on this data set are shown in Table 8.

From Figs. 3 and 4, one can see that the proposed algorithm is able to correctly detect the number of clusters on two synthetic data sets, however, the algorithm in [59] fails to detect the number of clusters on these two data sets.

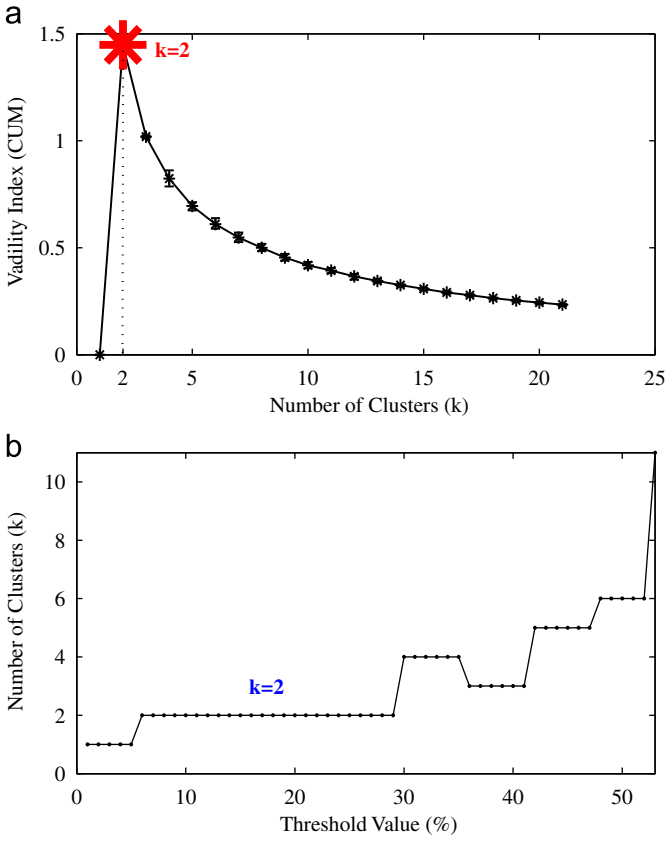

Fig. 7. The estimated number of clusters for the Voting data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

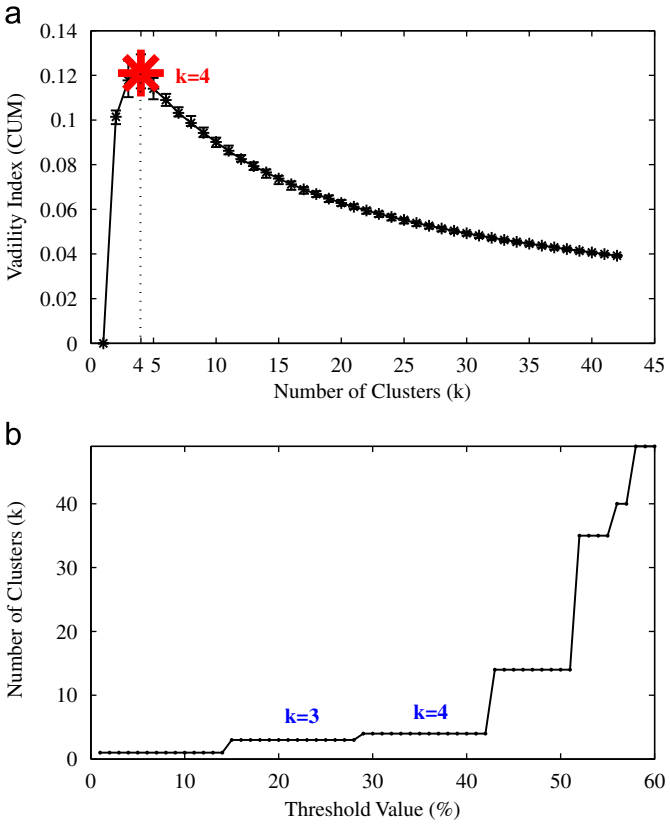

Fig. 8. The estimated number of clusters for the Car data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

Table 12
The summary results of three algorithms on the Voting data set.

Indices	Modified *k*-prototypes	*k*-Centers	Proposed algorithm
CUM	1.4478	0.9923	**1.4482**
ARI	0.5208	0.3555	**0.5340**

Table 13
The summary results of three algorithms on the Car data set.

Indices	Modified *k*-prototypes	*k*-Centers	Proposed algorithm
CUM	0.1140	0.1047	**0.1210**
ARI	0.0263	0.0215	**0.0323**

5.2. Numerical examples with real data sets

In this section, we have performed experiments with 10 different kinds of real data sets. These ten data sets are downloaded from the UCI Machine Learning Repository [64]. These representative data sets have two with numerical valued attributes, three with categorical valued attributes, and the others with a combination of numerical and categorical attributes. The data sets' characteristics are summarized in Table 9. In the following, we give the detailed information of these ten data sets and the corresponding experimental results, respectively.

Wine: This data set contains the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determines the quantities of 13 constituents found in each of the three types of wines. The attributes are, respectively, alcohol, malic acid, ash, magnesium, etc. The total number of instances in this data set is 178, i.e., 59 for class 1, 71 for class 2, and 48 for class 3. Fig. 5 shows the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm mentioned in [59]. The summary results of three different algorithms on this data set are shown in Table 10.

Breast Cancer: This data set was collected by Dr. William H. Wolberg at the University of Wisconsin Madison Hospitals. There are 699 records in this data set. Each record has nine attributes, which are graded on an interval scale from a normal state of 1–10, with 10 being the most abnormal state. In this database, 241 records are malignant and 458 records are benign. Fig. 6 shows the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm mentioned in [59]. The summary results of three different algorithms on this data set are shown in Table 11.

Voting: This UCI categorical data set gives the votes of each member of the U.S. House of Representatives of the 98th Congress on 16 key issues. It consists of 435 US House of Representative members' votes on 16 key votes (267 democrats and 168 republicans). Votes were numerically encoded as 0.5 for "yea", −0.5 for "nay" and 0 for unknown disposition, so that the voting record of each congressman is represented as a ternary-valued vector in R^{16}. Fig. 7 shows the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm mentioned in [59]. The summary results of three different algorithms on this data set are shown in Table 12.

Car: This data set evaluates cars based on their price and technical characteristics. This simple model was developed for educational purposes and is described in [63]. The data set has 1728 objects, each being described by six categorical attributes. The instances were classified into four classes, labeled

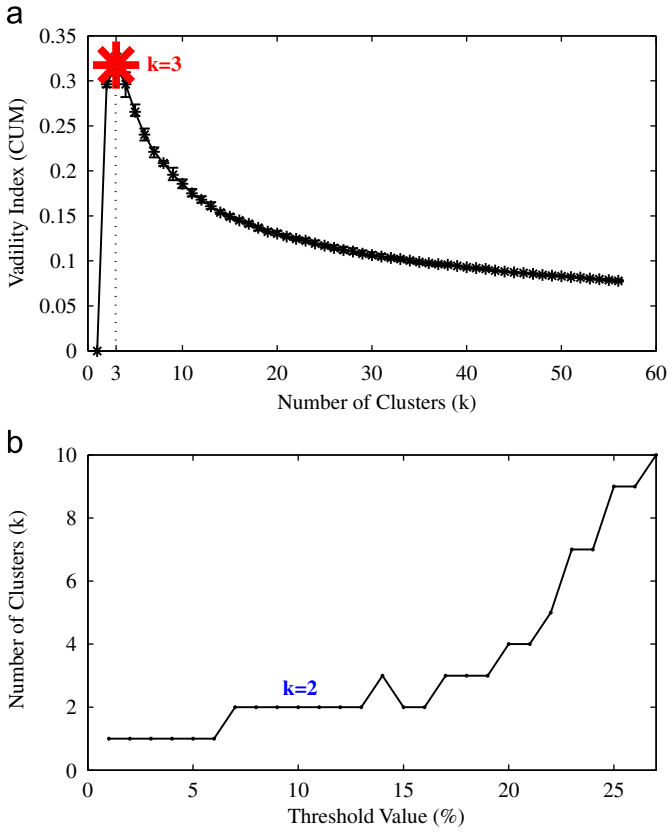

Fig. 9. The estimated number of clusters for the DNA data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

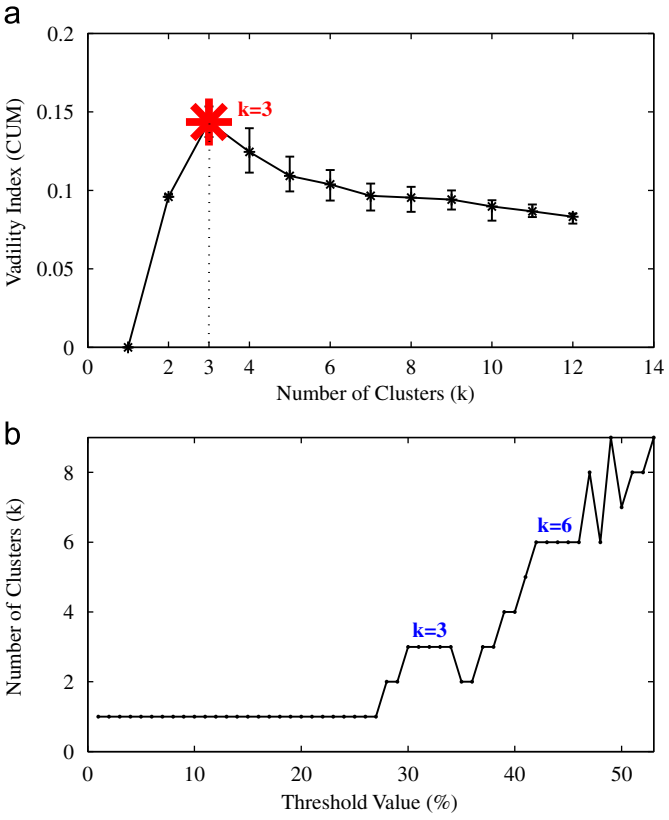

Fig. 10. The estimated number of clusters for the TAE data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

Table 14
The summary results of three algorithms on the DNA data set.

Indices	Modified *k*-prototypes	*k*-Centers	Proposed algorithm
CUM	0.2691	0.1483	**0.3175**
ARI	0.0179	**0.0414**	0.0240

Table 15
The summary results of three algorithms on the TAE data set.

Indices	Modified *k*-prototypes	*k*-Centers	Proposed algorithm
CUM	0.1160	0.0940	**0.1435**
ARI	0.0132	0.0154	**0.0256**

"unacc", "acc", "good" and "v-good". Fig. 8 shows the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm mentioned in [59]. The summary results of three different algorithms on this data set are shown in Table 13.

DNA: In this data set, each data point is a position in the middle of a window 60 DNA sequence elements. There is an intron/exon/neither field for each DNA sequence (which is not used for clustering). All of the 60 attributes are categorical and the data set contains 3190 data points (768 intron, 767 exon, and 1,655 neither). Fig. 9 shows the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm mentioned in [59]. The summary results of three different algorithms on this data set are shown in Table 14.

TAE: The data set consists of evaluations of teaching performance over three regular semesters and two summer semesters of 151 teaching assistant assignments at the Statistics Department of the University of Wisconsin-Madison. The scores were divided into three roughly equal-sized categories ("low", "medium", and "high") to form the class variable. It differs from the other data sets in that there are two categorical attributes with large numbers of categories. Fig. 10 shows the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm mentioned in [59]. The summary results of three different algorithms on this data set are shown in Table 15.

Heart: This data generated at the Cleveland Clinic, is a mixed data set with categorical and numerical features. Heart disease refers to the build-up of plaque on the coronary artery walls that restricts blood flow to the heart muscle, a condition that is termed "ischemia". The end result is a reduction or deprivation of the necessary oxygen supply to the heart muscle. The data set consists of 303 patient instances defined by 13 attributes. The data comes from two classes: people with no heart disease and people with different degrees (severity) of heart disease. We get the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm mentioned in [59] as plotted in Fig. 11. The summary results of three different algorithms on this data set are shown in Table 16.

Credit: The data set has 690 instances, each being described by six numerical and nine categorical attributes. The instances were classified into two classes, approved labeled as "+" and rejected labeled as "−". Fig. 12 shows the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm mentioned in [59]. The summary results of three different algorithms on this data set are shown in Table 17.

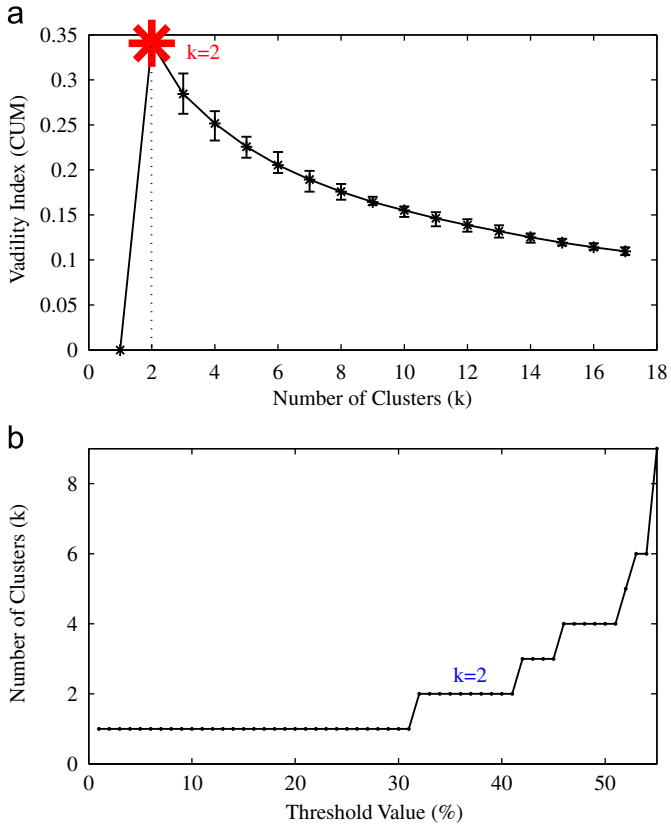

Fig. 11. The estimated number of clusters for the Heart data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

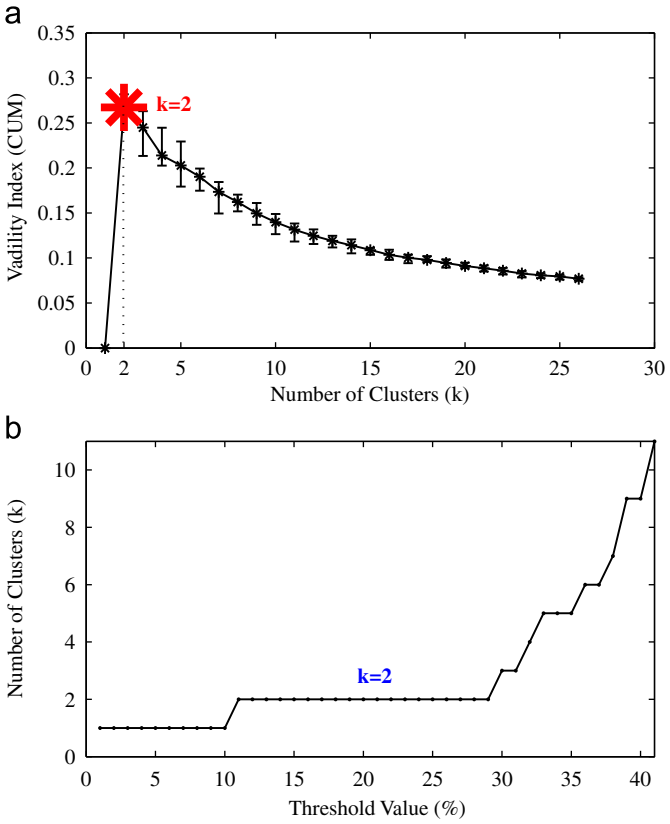

Fig. 12. The estimated number of clusters for the Credit data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

Table 16
The summary results of three algorithms on the Heart data set.

Indices	Modified *k*-prototypes	*k*-Centers	Proposed algorithm
CUM	**0.3406**	0.2017	**0.3406**
ARI	0.3303	0.1888	**0.3383**

Table 17
The summary results of three algorithms on the Credit data set.

Indices	Modified *k*-prototypes	*k*-Centers	Proposed algorithm
CUM	0.2658	0.1525	**0.2678**
ARI	0.2520	0.2323	**0.2585**

CMC: The data are taken from the 1987 National Indonesia Contraceptive Prevalence Survey. The samples are married women who were either not pregnant or did not know if they were pregnant at the time of the interview. The problem is to predict the current contraceptive method choice (no use, long-term methods, or short-term methods) of a woman based on her demographic and socio-economic characteristics. There are three classes, two numerical attributes, seven categorical attributes, and 1473 records. Fig. 13 shows the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm mentioned in [59]. The summary results of three different algorithms on this data set are shown in Table 18.

Adult: This data set was also from the UCI repository [64]. The dataset has 48 842 patterns of 15 attributes (eight categorical, six numerical, and one class attribute). The class attribute Salary indicates where the salary is over 50 K (>50 K) or 50 K or lower (≤ 50 K). Fig. 14 shows the estimated number of clusters for this data set by the proposed algorithm compared with the algorithm mentioned in [59]. Note that in order to show the variation tendency clearly, the numbers of clusters vary from 2 to 20 in this plot. The summary results of three different algorithms on this data set are shown in Table 19.

According to Figs. 5–14, it is clear that the numbers of clusters detected by the proposed algorithm are in agreement with the true numbers of these real data sets. However, the algorithm in [59] fails to detect the number of clusters on some real data sets, such as Breast Cancer, Car, DNA, TAE and CMC. As regards the clustering results shown in Tables 10–19, the proposed algorithm is superior to the other algorithms on the most data sets in terms of *CUM* and *ARI*.

5.3. Comparison in terms of time cost

In addition to the comparisons of the ability to detect the optimal number of clusters and obtain better clustering results, we have carried out time comparison between the proposed algorithm and the algorithm in [59]. The experiments are conducted on a PC with an Intel Pentium D processor (2.8 GHz) and 1 Gbyte memory running the Windows XP SP3 operating system. For statistical purposes, we ran these two algorithms 10 times and recorded the average number of the CPU time, respectively. For the algorithm in [59], it is difficult to set an appropriate step size of the similarity value threshold. Therefore, the similarity threshold varies from 0.01 to 1 with step-size 0.01 for all data sets used in this experiment. Once the algorithm starts producing

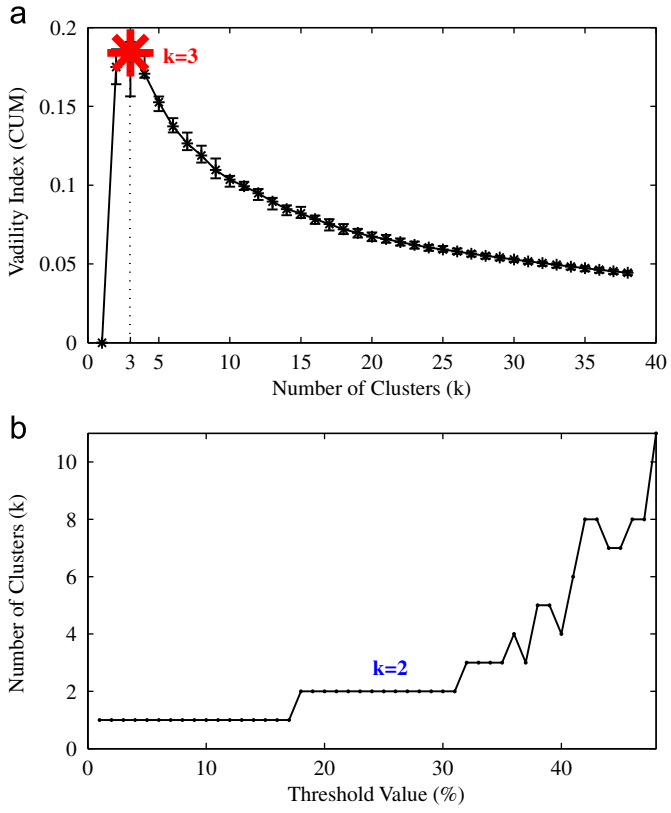

Fig. 13. The estimated number of clusters for the CMC data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

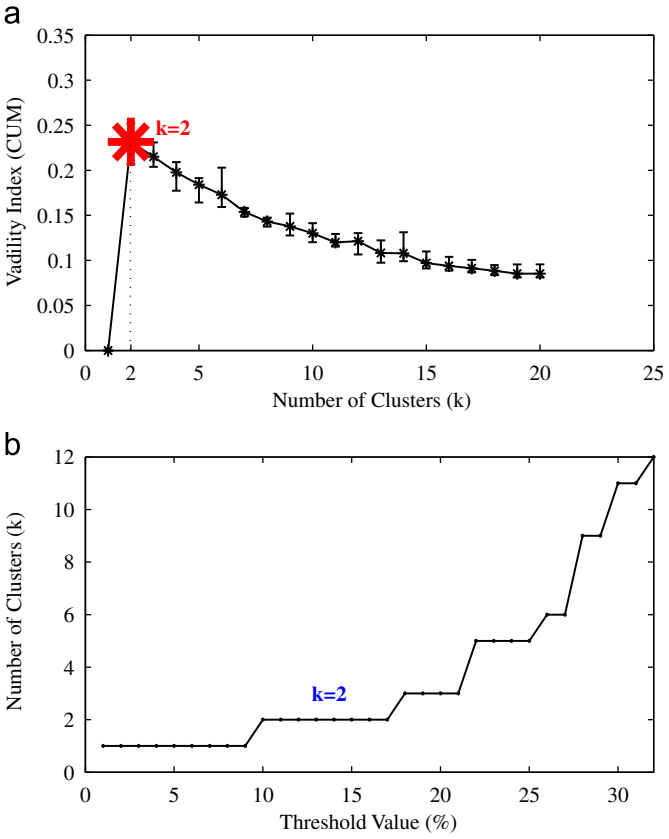

Fig. 14. The estimated number of clusters for the Adult data set (a) the proposed algorithm and (b) the algorithm mentioned in [59].

Table 18
The summary results of three algorithms on the CMC data set.

Indices	Modified *k*-prototypes	*k*-Centers	Proposed algorithm
CUM	0.1731	0.1513	**0.1839**
ARI	**0.0182**	0.0177	0.0167

Table 19
The summary results of three algorithms on the Adult data set.

Indices	Modified *k*-prototypes	*k*-Centers	Proposed algorithm
CUM	0.2170	0.1594	**0.2315**
ARI	0.1473	0.0937	**0.1742**

small interval length (L) of similarity threshold continuously ($L<2$), it will terminate. The comparisons of the CPU time on both synthetic and real data sets are shown in Table 20.

According to Table 20, our algorithm spends little time on four data sets, while the algorithm in [59] does the same on the other data sets. That is to say, there is no difference for these two algorithms in time consumption. However, the proposed algorithm can find the number of clusters and obtain better clustering results simultaneously, whereas the algorithm in [59] can only find the number of clusters. And the execution time of the algorithm in [59] depends on step size of the similarity threshold.

In summary, the experimental results performed on both synthetic and real data sets show the superiority and effectiveness of the proposed algorithm in detecting the correct number of clusters and obtaining better clustering results.

Table 20
The comparisons of execution time.

Data sets	Time consumption in second	
	The algorithm in [59]	Proposed algorithm
Ten-cluster	42.516	**34.25**
Student	**6.859**	27.672
Wine	3.672	**0.703**
Breast Cancer	**20.015**	21.984
Voting	**3.032**	5.656
Car	46.641	**34.531**
DNA	**214.063**	694.484
TAE	**1.172**	2.157
Heart	6.015	**4.093**
Credit	**3.703**	38.047
CMC	**57.719**	186.172
Adult	**613.735**	3657.484

6. Conclusions

The goal of this research is to develop a clustering algorithm for determining the optimal number of clusters for mixed data sets. In order to achieve this goal, a generalized mechanism for characterizing within-cluster entropy and between-cluster entropy and identifying the worst cluster in a mixed data set has been given by exploiting information entropy. To evaluate the clustering results, an effective cluster validity index has been defined by extending the category utility function. Based on the generalized mechanism, the cluster validity index and the

k-prototypes algorithm with a new dissimilarity measure, an algorithm has been developed to determine the number of clusters for mixed data sets. Experimental results on both synthetic and real data with mixed attributes show that the proposed algorithm is superior to the other algorithms both in detecting the number of clusters and in obtaining better clustering results.

Acknowledgments

The authors are very grateful to the anonymous reviewers and editor. Their many helpful and constructive comments and suggestions helped us significantly improve this work. This work was supported by the National Natural Science Foundation of China (Nos. 71031006,70971080,60970014), the Special Prophase Project on National Key Basic Research and Development Program of China (973) (No. 2011CB311805), the Foundation of Doctoral Program Research of Ministry of Education of China (No. 20101401110002) and the Key Problems in Science and Technology Project of Shanxi (No. 20110321027-01).

References

[1] J.W. Han, M. Kamber, Data Mining Concepts and Techniques, Morgan Kaufmann, San Francisco, 2001.
[2] A.K. Jain, M.N. Murty, P.J. Flynn, Data clustering: a review, ACM Computing Surveys 31 (3) (1999) 264–323.
[3] R. Xu, D. Wunsch II, Survey of clustering algorithms, IEEE Transactions on Neural Networks 16 (3) (2005) 645–678.
[4] A.K. Jain, Data clustering: 50 years beyond *k*-means, Pattern Recognition Letters 31 (8) (2010) 651–666.
[5] H.P. Kriegel, P. Kröger, A. Zimek, Clustering high-dimensional data: a survey on subspace clustering, pattern-based clustering and correlation clustering, ACM Transactions on Knowledge Discovery from Data 3 (1) (2009) 1–58.
[6] L. Bai, J.Y. Liang, C.Y. Dang, F.Y. Cao, A novel attribute weighting algorithm for clustering high-dimensional categorical data, Pattern Recognition 44 (12) (2011) 2843–2861.
[7] T.W. Liao, Clustering of time series data survey, Pattern Recognition 38 (11) (2005) 1857–1874.
[8] C.C. Aggarwal, J.W. Han, J.Y. Wang, P.S. Yu, A framework for clustering evolving data streams, in: Proceedings of the 29th VLDB Conference, Berlin, Germany, 2003.
[9] F.Y. Cao, J.Y. Liang, L. Bai, X.W. Zhao, C.Y. Dang, A framework for clustering categorical time-evolving data, IEEE Transactions on Fuzzy Systems 18 (5) (2010) 872–882.
[10] L. Hunt, M. Jorgensen, Clustering mixed data, WIREs Data Mining and Knowledge Discovery 1 (4) (2011) 352–361.
[11] V. Estivill-Castro, J. Yang, A fast and robust general purpose clustering algorithm, in: Proceeding of 6th Pacific Rim International Conference Artificial Intelligence, Melbourne, Australia, 2000, pp. 208–218.
[12] A. Ahmad, L. Dey, A *k*-mean clustering algorithm for mixed numeric and categorical data, Data & Knowledge Engineering 63 (2) (2007) 503–527.
[13] Z.X. Huang, Extensions to the *k*-means algorithm for clustering large data sets with categorical values, Data Mining and Knowledge Discovery 2 (3) (1998) 283–304.
[14] J.B. MacQueen, Some methods for classification and analysis of multivariate observations, in: Proceeding of the 5th Berkeley Symposium on Mathematical Statistics and Probability, 1967, pp. 281–297.
[15] F. Höppner, F. Klawonn, R. Kruse, Fuzzy Cluster Analysis: Methods for Classification, Data Analysis, and Image Recognition, Wiley, New York, 1999.
[16] Z.X. Huang, A fast clustering algorithm to cluster very large categorical data sets in data mining, in: Proceeding of the SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, 1997, pp. 1–8.
[17] L. Kaufman, P. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis, Wiley, 1990.
[18] T.K. Xiong, S.R. Wang, A. Mayers, E. Monga, DHCC: Divisive hierarchical clustering of categorical data, Data Mining and Knowledge Discovery, doi:10.1007/s10618-011-0221-2 2011.
[19] C. Li, G. Biswas, Unsupervised learning with mixed numeric and nominal data, IEEE Transactions on Knowledge and Data Engineering 14 (4) (2002) 673–690.
[20] C.C. Hsu, C.L. Chen, Y.W. Su, Hierarchical clustering of mixed data based on distance hierarchy, Information Sciences 177 (20) (2007) 4474–4492.
[21] S. Guha, R. Rastogi, K. Shim, CURE: An efficient clustering algorithm for large databases, in: Proceeding of ACM SIGMOD International Conference Management of Data, 1998, pp. 73–84.
[22] G. Karypis, E. Han, V. Kumar, Chameleon: hierarchical clustering using dynamic modeling, IEEE Computer 32 (8) (1999) 68–75.
[23] T. Zhang, R. Ramakrishnan, M. Livny, BIRCH: an efficient data clustering method for very large databases, in: Proceeding of ACM SIGMOD International Conference Management of Data, 1996, pp. 103–114.
[24] S. Guha, R. Rastogi, K. Shim, ROCK: a robust clustering algorithm for categorical attributes, Information Systems 25 (5) (2000) 345–366.
[25] B. Mirkin, Choosing the number of clusters, WIREs Data Mining and Knowledge Discovery 1 (3) (2011) 252–260.
[26] H.J. Sun, S.R. Wang, Q.S. Jiang, FCM-based model selection algorithms for determining the number of clusters, Pattern Recognition 37 (10) (2004) 2027–2037.
[27] R. Kothari, D. Pitts, On finding the number of clusters, Pattern Recognition Letters 20 (4) (1999) 405–416.
[28] M.J. Li, M.K. Ng, Y. Cheung, Z.X. Huang, Agglomerative fuzzy *k*-means clustering algorithm with selection of number of clusters, IEEE Transactions on Knowledge and Data Engineering 20 (11) (2008) 1519–1534.
[29] Y. Leung, J.S. Zhang, Z.B. Xu, Clustering by scale-space filtering, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (12) (2000) 1394–1410.
[30] S. Bandyopadhyay, U. Maulik, Genetic clustering for automatic evolution of clusters and application to image classification, Pattern Recognition 35 (6) (2002) 1197–1208.
[31] S. Bandyopadhyay, S. Saha, A point symmetry-based clustering technique for automatic evolution of clusters, IEEE Transactions on Knowledge and Data Engineering 20 (11) (2008) 1441–1457.
[32] S. Bandyopadhyay, Genetic algorithms for clustering and fuzzy clustering, WIREs Data Mining and Knowledge Discovery 1 (6) (2011) 524–531.
[33] C.A. Sugar, G.M. James, Finding the number of clusters in a data set: an information theoretic approach, Journal of the American Statistical Association 98 (463) (2003) 750–763.
[34] M. Aghagolzadeh, H. Soltanian-Zadeh, B.N. Araabi, A. Aghagolzadeh, Finding the number of clusters in a dataset using an information theoretic hierarchical algorithm, in: Proceedings of the 13th IEEE International Conference on Electronics, Circuits and Systems, 2006, pp. 1336–1339.
[35] L. Bai, J.Y. Liang, C.Y. Dang, An initialization method to simultaneously find initial cluster centers and the number of clusters for clustering categorical data, Knowledge-Based Systems 24 (6) (2011) 785–795.
[36] K.K. Chen, L. Liu, The best *k* for entropy-based categorical clustering, in: Proceeding of the 17th International Conference on Scientific and Statistical Database Management, 2005.
[37] H. Yan, K.K. Chen, L. Liu, J. Bae, Determining the best *k* for clustering transactional datasets: a coverage density-based approach, Data & Knowledge Engineering 68 (1) (2009) 28–48.
[38] R. Jensen, Q. Shen, Fuzzy-rough sets for descriptive dimensionality reduction, in: Proceeding of the 2002 IEEE International Conference on Fuzzy Systems, 2002, pp. 29–34.
[39] C.E. Shannon, A mathematical theory of communication, Bell Systems Technical Journal 27 (3-4) (1948) 379–423.
[40] D. Barbara, Y. Li, J. Couto, Coolcat: an entropy-based algorithm for categorical clustering, in: Proceeding of the 2002 ACM CIKM International Conference on Information and Knowledge Management, 2002, pp. 582–589.
[41] Z.Y. He, S.C. Deng, X.F. Xu, An optimization model for outlier detection in categorical data. in: Lecture Notes in Computer Science, vol. 3644, 2005, pp. 400–409.
[42] I. Düntsch, G. Gediga, Uncertainty measures of rough set prediction, Artificial Intelligence 106 (1) (1998) 109–137.
[43] A. Renyi, On measures of entropy and information, in: Proceeding of the 4th Berkeley Symposium on Mathematics of Statistics and Probability, 1961, pp. 547–561.
[44] E. Parzen, On the estimation of a probability density function and the mode, Annals of Mathematical Statistics 33 (3) (1962) 1065–1076.
[45] R. Jenssen, T. Eltoft, D. Erdogmus, J.C. Principe, Some equivalences between kernel methods and information theoretic methods, Journal of VLSI Signal Processing Systems 49 (1–2) (2006) 49–65.
[46] E. Gokcay, J.C. Principe, Information theoretic clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2) (2002) 158–171.
[47] J.Y. Liang, K.S. Chin, C.Y. Dang, C.M. Yam Richard, A new method for measuring uncertainly and fuzziness in rough set theory, International Journal of General Systems 31 (4) (2002) 331–342.
[48] Y.H. Qian, J.Y. Liang, W. Pedrycz, C.Y. Dang, Positive approximation: an accelerator for attribute reduction in rough set theory, Artificial Intelligence 174 (9–10) (2010) 597–618.
[49] Y.H. Qian, J.Y. Liang, D.Y. Li, H.Y. Zhang, C. Y Dang, Measures for evaluating the decision performance of a decision table in rough set theory, Information Sciences 178 (1) (2008) 181–202.
[50] J.Y. Liang, D.Y. Li, Uncertainty and Knowledge Acquisition in Information Systems, Science Press, Beijing, China, 2005.
[51] J.Y. Liang, Z.Z. Shi, D.Y. Li, M.J. Wierman, The information entropy, rough entropy and knowledge granulation in incomplete information system, International Journal of General Systems 35 (6) (2006) 641–654.
[52] M. Halkidi, M. Vazirgiannis, A density-based cluster validity approach using multi-representatives, Pattern Recognition Letters 29 (6) (2008) 773–786.
[53] M. Rezaee, B. Lelieveldt, J. Reiber, A new cluster validity index for the fuzzy c-mean, Pattern Recognition Letters 19 (3-4) (1998) 237–346.
[54] W.N. Wang, Y.J. Zhang, On fuzzy cluster validity indices, Fuzzy Sets and Systems 158 (19) (2007) 2095–2117.

[55] M.A. Gluck, J.E. Corter, Information, uncertainty, and the utility of categories, in: Proceeding of the 7th Annual Conference of the Cognitive Science Society, Lawrence Erlbaum Associates, Irvine, CA, 1985, pp. 283–287.
[56] D.H. Fisher, Knowledge acquisition via incremental conceptual clustering, Machine Learning 2 (2) (1987) 139–172.
[57] K. McKusick, K. Thompson, COBWEB/3: A Portable Implementation, Technical Report FIA-90-6-18-2, NASA Ames Research Center, 1990.
[58] B. Mirkin, Reinterpreting the category utility function, Machine Learning 45 (2) (2001) 219–228.
[59] J. Al-Shaqsi, W.J. Wang, A clustering ensemble method for clustering mixed data, in: The 2010 International Joint Conference on Neural Networks, 2010.
[60] W.D. Zhao, W.H. Dai, C.B. Tang, K-centers algorithm for clustering mixed type data, in: Proceedings of the 11th Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2007, pp. 1140–1147.
[61] J.C. Bezdek, Pattern Recognition in Handbook of Fuzzy Computation, IOP Publishing Limited, Boston, New York, 1998. (Chapter F6).
[62] L. Hubert, P. Arabie, Comparing partitions, Journal of Classification 2 (1) (1985) 193–218.
[63] M. Bohanec, V. Rajkovic, Knowledge acquisition and explanation for multi-attribute decision making, in: Proceeding of the 8th International Workshop on Expert Systems and Their Applications, Avignon, France, 1988, pp. 59–78.
[64] UCI Machine Learning Repository 〈http://www.ics.uci.edu/mlearn/MLRepository.html〉, 2011.

Jiye Liang is a professor of School of Computer and Information Technology and Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education at Shanxi University. He received his M.S. degree and Ph.D. degree from Xi'an Jiaotong University in 1990 and 2001, respectively. His current research interests include computational intelligence, granular computing, data mining and knowledge discovery. He has published more than 100 journal paper in his research fields.

Xingwang Zhao is a teaching assistant in the School of Computer and Information Technology in Shanxi University. He received his M.S. degree from Shanxi University in 2011. His research interests are in the areas of data mining and machine learning.

Deyu Li is a professor of School of Computer and Information Technology of Shanxi University. He received his M.S. degree from Shanxi University in 1998, and his Ph.D. degree from Xi'an Jiaotong University in 2002. His current research interests include rough set theory, granular computing, data mining and knowledge discovery.

Fuyuan Cao received his M.S. and Ph.D. degrees in Computer Science from Shanxi University in 2004 and 2010, respectively. Now, he is an Associate Professor with the School of Computer and Information Technology in Shanxi University. His research interests include data mining and machine learning.

Chuangyin Dang received a Ph.D. degree in operations research/economics from the University of Tilburg, The Netherlands, in 1991, a M.S. degree in applied mathematics from Xidian University, China, in 1986, and a B.S. degree in computational mathematics from Shanxi University, China, in 1983. He is Associate Professor at the City University of Hong Kong. He is best known for the development of the D1-triangulation of the Euclidean space and the simplicial method for integer programming. His current research interests include computational intelligence, optimization theory and techniques, applied general equilibrium modeling and computation. He is a senior member of IEEE and a member of INFORS and MPS.